Cloudlet-Aware Mobile Content Delivery in Wireless Urban Communication Networks

Hassan Sinky and Bechir Hamdaoui
Oregon State University, Emails: sinkyh@oregonstate.edu, hamdaoui@eecs.oregonstate.edu

***Abstract*—LinkNYC is a first-of-its-kind urban communications network aiming to replace all payphones in the five boroughs of New York City with kiosk-like structures providing free super fast gigabit Wi-Fi to everyone. This work proposes and investigates the applicability of shifting LinkNYC from a traditional IP network to a content-centric network by upgrading all or a subset of their kiosks with standalone cloudlets, which cache content as it disseminates throughout the network, and content delivery cloudlets which are geographically distributed throughout the boroughs and store popular internet content. With this shift content is brought much closer to the end user than traditional methods which is essential in highly mobile environments. Analysis shows that adopting multiple content delivery cloudlets dramatically improves overall network performance and stability. Finally, given a cloudlet-aware path a mobile content delivery scheme is designed to offset service continuity issues that are amplified when a mobile user encounters multiple cloudlets with intermittent connectivity. Results show an overall improvement in a mobile user's throughput, response time and cache hit percentage.**

I. Introduction

The proliferation of internet devices has resulted in efforts to integrate various wireless access technologies for improved performance, increased services and inter-connectivity of end users. The recent growth in data demand has prompted researchers to come up with new wireless techniques (e.g., MIMO [1], [2], cooperative communication [3], femtocells [4], etc.) and develop new technologies (e.g., cognitive radio [5]–[7], LTE [8], etc.) to be able to meet this high demand. This integration allows for large geographic locations to be serviced providing millions of end users with continuous connectivity and optimal quality of experience (QoE). However, the world has seen unprecedented urban population growth over the years. In fact, the number of urban residents has increased by nearly 60 million a year. By 2050, it is estimated that 70% of the world's population will be living in cities[1]. Urban communication networks and content delivery networks have been introduced to leverage these technologies to better service cities and users alike. Content delivery networks are designed to improve overall network performance by bringing data closer to the geographical locations of users. Urban communication networks have evolved over the years to address urban challenges through the use of information, communication technology and the Internet. Building such a network infrastructure capable of adequately servicing urban locations has become increasingly difficult due to the shear number of internet devices and users.

Traditionally, content delivery nodes are geographically distributed throughout the world servicing different regions. In large networks such as LinkNYC the same content may be requested by multiple users resulting in the content traversing the entire network multiple times to and from a remote content delivery node hosting the content. This work, however, proposes and analyzes the placement of content delivery cloudlets within LinkNYC's infrastructure to bring content closer to consumers. This is especially helpful for mobile LinkNYC users naturally experiencing intermittent connectivity as they associate with different cloudlets across a path. Thus, a cloudlet-aware mobile content delivery scheme is proposed to address mobile service continuity issues. Contributions of this work are:

- Performance analysis of the proposed shift of LinkNYC's infrastructure from a traditional communications network to a content-centric network.
- Establishes that LinkNYC's communications network vastly improves with not only a content-centric shift but also the placement of multiple content delivery cloudlets geographically distributed throughout LinkNYC.
- Designs and evaluates a cloudlet-aware mobile content delivery scheme for mobile users undergoing frequent handoffs and intermittent connectivity within LinkNYC.
- To our knowledge, this work is the first to analyze and leverage LinkNYC's urban communications network through content delivery cloudlets improving overall network performance and stability as well as mobile content delivery and service continuity.

The rest of this paper is organized as follows. In Section II a content-centric LinkNYC infrastructure is introduced and analyzed. A cloudlet-aware mobile content delivery scheme is designed and evaluated in Section III. Finally, the article is concluded in Section IV.

II. Content-Centric Urban Communication Networks

Coupling urban communication networks with content-centric and delivery principles greatly benefit content producers, consumers and the cities they reside in. Improving their infrastructure using practical approaches to provide more reliable and responsive communications can assist in the technology's overall success. The underlying concept behind content-centric communication networks is to allow a consumer to focus on the desired named content rather than referencing the physical

This work was supported in part by the US National Science Foundation (NSF) under NSF award CNS-1162296.

[1]World population data sheet: http://www.prb.org/

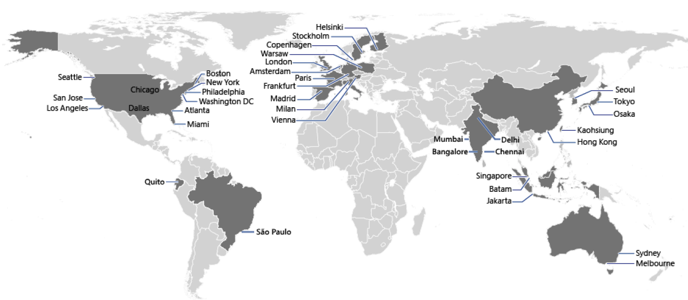


Figure 1: Microsoft Azure CDN point of presence locations

Borough	# Payphones	Avg. distance
Manhattan	3409	43.2 m
Queens	1042	136.8 m
Brooklyn	1004	150.8 m
Bronx	591	125.5 m
Staten Island	51	606 m
Total	6097	212.5 m

Table I: Payphones in the five boroughs of NYC

location or named hosts (IP) where that content is stored. This shift is a product of empirical research resulting in the fact that the vast majority of Internet usage involves data being disseminated from a source to multiple users. The potential benefits of a content-centric adoption include in-network caching to reduce congestion, improved delivery speeds, simpler network configuration and network security at the data level [9]. This paper combines content-centric and content delivery principles to improve LinkNYC performance and reliability.

A. Content Delivery Networks or CDNs

The principle behind CDNs is to bring data as close to the geographic location of the user as possible to improve overall network performance. This helps eliminate the need to traverse the Internet for content which reduces infrastructure and bandwidth costs while improving network robustness and QoE. For instance, Microsoft Azure's content delivery network consists of 36 points of presence locations[2] distributed throughout the world as shown in Figure 1. However, in addition to being geographically limited, CDN nodes are generally placed at the Internet edges over multiple backbones servicing different regions remotely. Although the concept of bringing content closer to the consumer through caching copies in various geographic locations improves overall network performance, mobile users in large urban communication networks such as LinkNYC naturally endure additional latency due to increased mobility, congestion and hops traversed within the network. This can be improved by placing content even closer to the requesting consumer through content-centric networking and delivery principles and is discussed next.

B. LinkNYC

In November 2014 LinkNYC announced a project plan to provide a first-of-its-kind communications network offering super fast free gigabit Wi-Fi to everyone in New York City through the replacement of thousands of payphones with kiosk-like structures called **Links** with deployment underway beginning January 2016. Once completed, LinkNYC will be the largest and fastest free public Wi-Fi network in the world. The Links are designed as an update to the standard phone booth and act as Wi-Fi hotspots while also providing basic services such as advertisements, free phone calls, device charging, touchscreen for Internet browsing to access city services, maps and directions. Revenue generated by the Links, through kiosk Ads that are displayed on 55 inch displays, is used to maintain the LinkNYC infrastructure. Each Link is equipped with 802.11ac Wi-Fi technology yielding download and upload speeds of 300 and 320 Mbps respectively with an average latency of 5 ms and coverage area of up to 45 meters depending on location. This promising urban communications network provides cities with revenue and analytics while offering consumers free, continuous and reliable connectivity.

In order to leverage content-centric and content delivery networking with LinkNYC's communications infrastructure this work proposes an architectural addition of cloudlets to better service end users. Cloudlets are small-scale cloud datacenters that aim to bring data closer to mobile users and are typically located near the edge of the Internet [10]. Upgrading LinkNYC's Links with cloudlets offers a different dimension to content delivery networking. Naturally, the number of potential cloudlets depends on the number of currently installed payphone locations[3] summarized in Table I. Manhattan is the most dense of the five boroughs with 3,409 payphones and an average distance between them of 43 meters. Unlike traditional CDNs, where a limited number of remote servers or nodes are distributed throughout the world, the proposed approach geographically places cloudlets within LinkNYC's network. This work proposes the placement of cloudlets in all or a subset of LinkNYC Links. Specific Links are selected as content delivery cloudlets or producers of content and are equipped with an $L2$ storage cache that is much larger than the $L1$ storage cache available on other cloudlets. In order to decide which cloudlets will provide content delivery services a hierarchical clustering technique is used.

C. Hierarchical Clustering

As consumers become increasingly mobile the placement of content delivery cloudlets within a large network is crucial. Specifically, mobile users that undergo frequent handoffs as they move across a path results in transport layer issues that reduce service continuity and overall QoE [11]. Placing a single content delivery cloudlet or content producer within the network is insufficient to meet the demand of the mobile consumer. Having content readily available in multiple nearby content delivery cloudlets helps avoid the additional costs of

[2]Figure 1 provided by Microsoft Azure: https://azure.microsoft.com/en-us/documentation/articles/cdn-pop-locations

[3]NYC Open Data: https://nycopendata.socrata.com

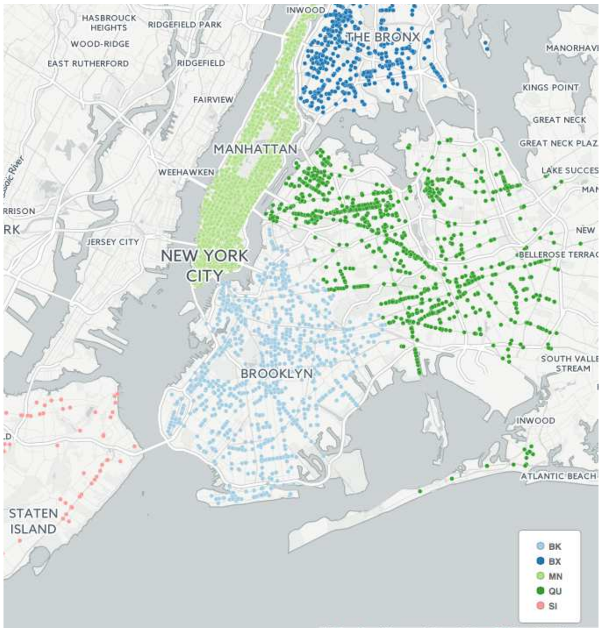


Figure 2: Geographic locations of payphones in NYC

traversing the network for the content. In an effort to analyze LinkNYC's network and decide for the placement of content delivery cloudlets a hierarchical clustering technique is applied to the borough topologies.

LinkNYC cloudlets are categorized based on their borough as shown in Figure 2. Since physical characteristics of New York City's payphone backhaul connectivity are unknown practical assumptions are made. First, we assume cloudlets are physically connected by fiber optic cables to their nearest neighbor. Given a particular NYC borough, i.e. Brooklyn, a euclidean minimum spanning tree (EMST) is constructed as shown in Figure 3 using Prim's algorithm where edge weights equal the geographic distance between cloudlets. This results in a network topology with $N-1$ edges where N is the number of cloudlets in a particular borough. Second, content delivery cloudlets are geographically distributed throughout the borough network based on an edge-betweenness hierarchical clustering technique. This technique, known as the Girvan–Newman algorithm, progressively removes edges from the original topology that are least central to clusters to form network communities [12]. Edges are ranked based on the number of shortest path combinations that run through them. Higher ranked edges are assumed to be most "between" communities. The edge with the highest edge-betweenness is removed. Edge-betweenness is then recalculated for the edges that are affected by the removal. This process is repeated until no edges remain resulting in a set of communities. As shown in Figure 4, the number of community clusters created for Brooklyn's topology is 24.

D. Content Delivery Cloudlet Placement

Given the resulting cloudlet communities from Section II-C one node is selected per community as the content delivery or producer cloudlet which stores content in its $L2$ cache. The

Borough	**Clusters**	**CDN %**
Manhattan	64	1.9%
Queens	37	3.6%
Brooklyn	24	2.4%
Bronx	30	5.1%
Staten Island	9	17.6%
Total	171	2.8%

Table II: Content delivery cloudlets in each borough

producer cloudlet is selected based on its distance from the remaining cloudlets within its respective cluster. That is, the node with the minimum sum of shortest paths to the other nodes is selected as the content delivery cloudlet. This ensures content is placed as close to the geographic location of potential consumers within a cluster as possible. Table II shows the results after applying this technique to the remaining boroughs. LinkNYC's overall communications network yields 171 total content delivery cloudlets which makes up 2.8% of the total number of cloudlets within New York City.

Figure 3: Brooklyn's LinkNYC Network

E. Borough Analysis

Different Internet architectures have been introduced for content-centric networking which shifts from the standard IP data packets to Named-Data Networking (NDN). NDN is a future Internet architecture focusing on a content-centric Internet as opposed to today's host-centric network architecture [9], [13], [14]. These architectures rely on named-content within the Internet to route and direct the flow of data within a network. In these architectures the content is the focus rather than the physical location where the content is stored. `ndnSIM` is an NDN simulator based on NS-3 and was used to analyze the proposed urban communications network infrastructure.

`ndnSIM` consists of content consumers and producers. Consumers generate interest requests for specific content chunks whereas producers respond to interest requests with data packets. Every node is equipped with a Content Store (CS), Pending Interest Table (PIT), and Forwarding Information Base (FIB). When interest packets are received it is first placed into the

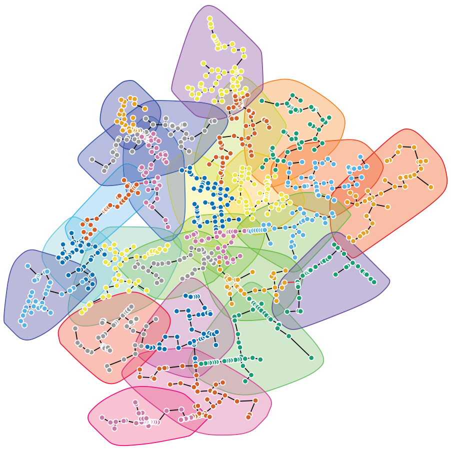


Figure 4: Clustering of Brooklyn's LinkNYC network

Parameter	Value
Producers	24
Consumers	60
L1 Cache	1 GB
L2 Cache	1 GB
Requests	100 per seconds
Max SeqNo	3000
MTU	1400 bytes
Link Rate	10 Mbps
Latency	5 ms

Table III: Brooklyn Simulation parameters

PIT and the CS is checked for data correlating to this interest. If there is a match the interest request is discarded and the corresponding data packet from the CS is returned. Otherwise the interest packet is forwarded based on information in the FIB. Incoming data packets corresponding to pending interests in the PIT are stored in the node's CS. Otherwise, the data packet is dropped.

For this analysis a comparison is made between traditional, random and cluster-based content delivery cloudlet placements in Brooklyn which comprises of 1004 nodes. The traditional approach randomly selects and places a single content delivery cloudlet to service this specific borough. This is done with and without intermediate cloudlets caching content to emphasize the improvement in performance when shifting to the content-centric paradigm. The random approach randomly selects 24 content delivery cloudlets whereas cluster-based selects one cloudlet per cluster as mentioned in Section II-C. For each simulation 60 consumers are randomly selected and initiate requests for the same content at different time intervals between 0–60 seconds. Results of this analysis are shown in Figure 5.

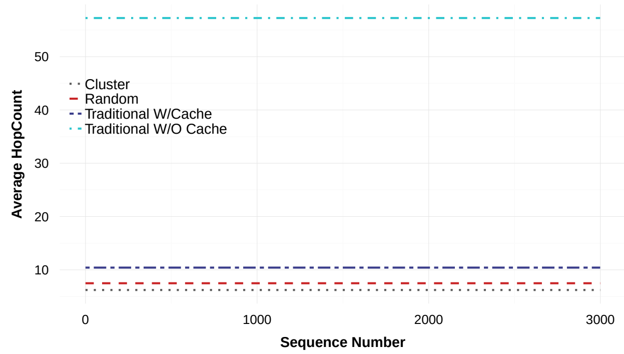
Figure 5a shows the network at equilibrium when all consumers have received the final sequence number. This is a promising result as the difference between traditional content delivery placement with and without caching is quite dramatic and emphasizes the improvement of a content-centric shift within LinknNYC's network. The average number of hops traveled by the traditional method with caching is slightly over 10 hops compared to over 50 hops for the traditional method without caching. Without caching, content is forced to traverse the entire network to reach the regional content delivery cloudlet resulting in a much higher hop count. In addition, randomly distributing 24 content delivery cloudlets improves the average number of hops to a little under 8. Finally, the cluster-based approach achieves better results yielding on average around 6 hops. Figure 5b shows the average hop count midway through the simulation. This is an important distinction to make as it highlights the practical scenarios where, since consumers request content or join/ leave the network at different time intervals, higher sequence numbers have yet to be requested and fully disseminate throughout the network. This results in high sequence numbers not being available in cloudlet $L1$ caches causing intermediate cloudlets to request the content from their neighbors and in turn requiring more hops to traverse.

Figures 5d and 5c show the total number of cache hits and ratio of hits to misses respectively at the content delivery cloudlets over time. A cache hit occurs when data corresponding to an interest packet is fulfilled from the in-network local cache, $L1$ or $L2$, of the cloudlets rather than from the producer application. As more users join the network the cluster-based approach achieves a higher number of cache hits as well as a higher hit to miss ratio for requested content. This is attributed to the cluster-based placement of the content delivery cloudlets as opposed to a random selection. That is, the likelihood that the cluster-based approach would place cloudlets closer to potential consumers is higher than other approaches. Eventually when equilibrium is reached, the cache hit to miss ratios converge for random and cluster-based approaches as higher sequence numbers have fully disseminated the network resulting in higher cache hits. However, traditional methods yield poor results as they experience low cache hit percentages.

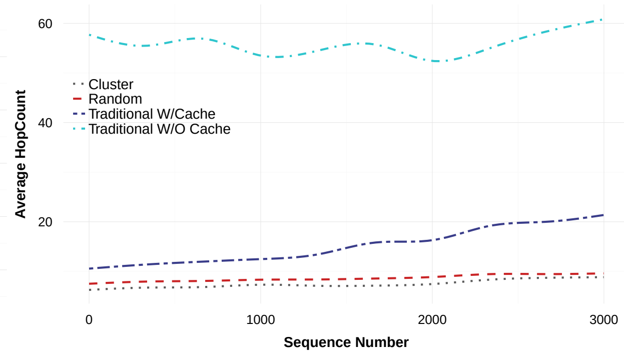
This analysis shows that LinkNYC can benefit greatly with not only the adoption of a content-centric infrastructure (traditional with caching) but also with the incorporation of multiple content delivery cloudlets within its urban communications network. Overall a cluster-based placement approach exhibits promising and more stable results yielding lower hop counts and achieving on average a higher rate of cache hits which in turn improves content delivery speeds. In a mobile environment, where network conditions and service continuity issues are amplified due to mobility, network stability and content delivery speeds are essential to a user's QoE. Section III introduces a cloudlet-aware mobile content delivery scheme for LinkNYC's communications network given a user's mobility within a cluster serviced by a content delivery cloudlet.

III. LinkNYC Mobile Content Delivery

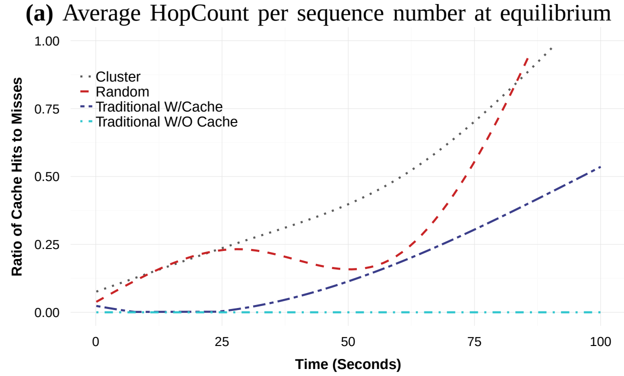
It is clear from Figure 2 that New York City is densely equipped with thousands of potential cloudlets within an area populated with multiple mobile users. Within this environment mobile users naturally experience frequent handoffs during



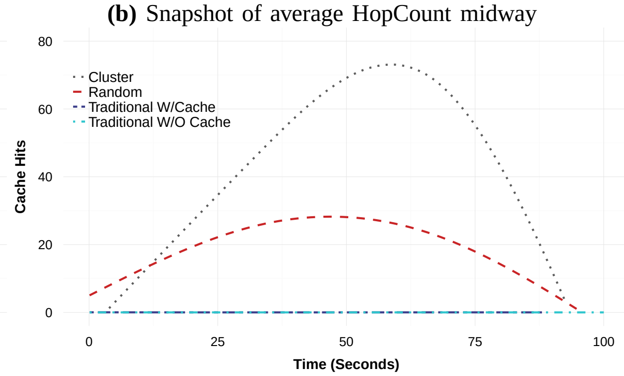
(a) Average HopCount per sequence number at equilibrium



(b) Snapshot of average HopCount midway



(c) Content delivery cloudlet cache hit to miss ratio



(d) Total cache hits on content delivery cloudlets over time

Figure 5: Content-centric LinkNYC performance results

a connection lifetime resulting in intermittent connectivity which is detrimental for mobile service continuity and overall QoE [11]. Content interest requests must go through a mobile user's point of attachment (PoA) or currently connected to cloudlet. This real world scenario risks frequent disconnects and disruptions in a user's service especially near the edges of a cloudlet's coverage area. In this case, even though a handoff may be imminent, interest requests must still be requested through the PoA. This causes potential packet losses, increased response times and re-transmissions. Although the placement of content delivery cloudlets improves overall network performance, the physical characteristics and consequences of a mobile user are inevitable. This requires the need to counter these service continuity issues and disruptions by prefetching content on the requesting user's expected path.

A. Design

This work uses the architecture from Figure 6 to address the mobile issue by prefetching content on candidate cloudlets located on a mobile user's path. A cloudlet-aware GPS is assumed to provide a mobile user with a path containing multiple cloudlets. In addition, each content delivery cloudlet within a cluster is responsible for maintaining critical cloudlet information such as location, throughput, coverage area and expected traveling speed. This information is shared between the content delivery cloudlets. A cloudlet-aware path P_N, where N is the number of cloudlets on a mobile user's path, consists of a 4-tuple, (C_i, R_i, d_i, s_i), representing cloudlet C_i,

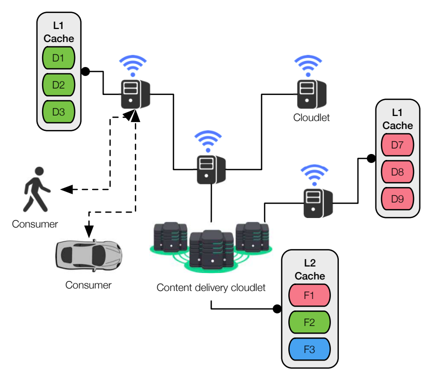


Figure 6: Mobile content delivery cloudlet architecture.

where R_i and d_i are C_i's throughput and coverage area, and s_i is the expected speed traveled within C_i's coverage area. In order to prefetch chunks of content Algorithm 1 is applied by a mobile user as it moves within a cluster of cloudlets.

First, a mobile user obtains a cloudlet-aware path, $P_N = (C_1, R_1, d_1, s_1), ..., (C_N, R_N, d_N, s_N)$, through a central server where cloudlet details are maintained while also continuously monitoring its connection rate and speed within the current PoA cloudlet coverage area. Second, a content specific manifest file, which contains content details such as file size, is requested from the cluster content delivery cloudlet. Once received, the mobile user parses the manifest file to acquire the content size and in turn the maximum sequence number,

Parameter	Value
L1 Cache	1 GB
L2 Cache	2 GB
File size	200 MB
Requests	1800 per second
MTU	1449 bytes
R_i	54 Mbps (802.11a)
d_i	20 meters
s_i	1-2 m/s
Δ_0	20 meters

Table IV: Simulation parameters

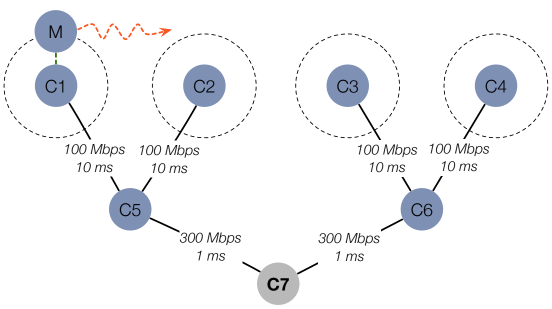


Figure 7: Simulation cluster topology.

Algorithm 1 Mobile Content Delivery

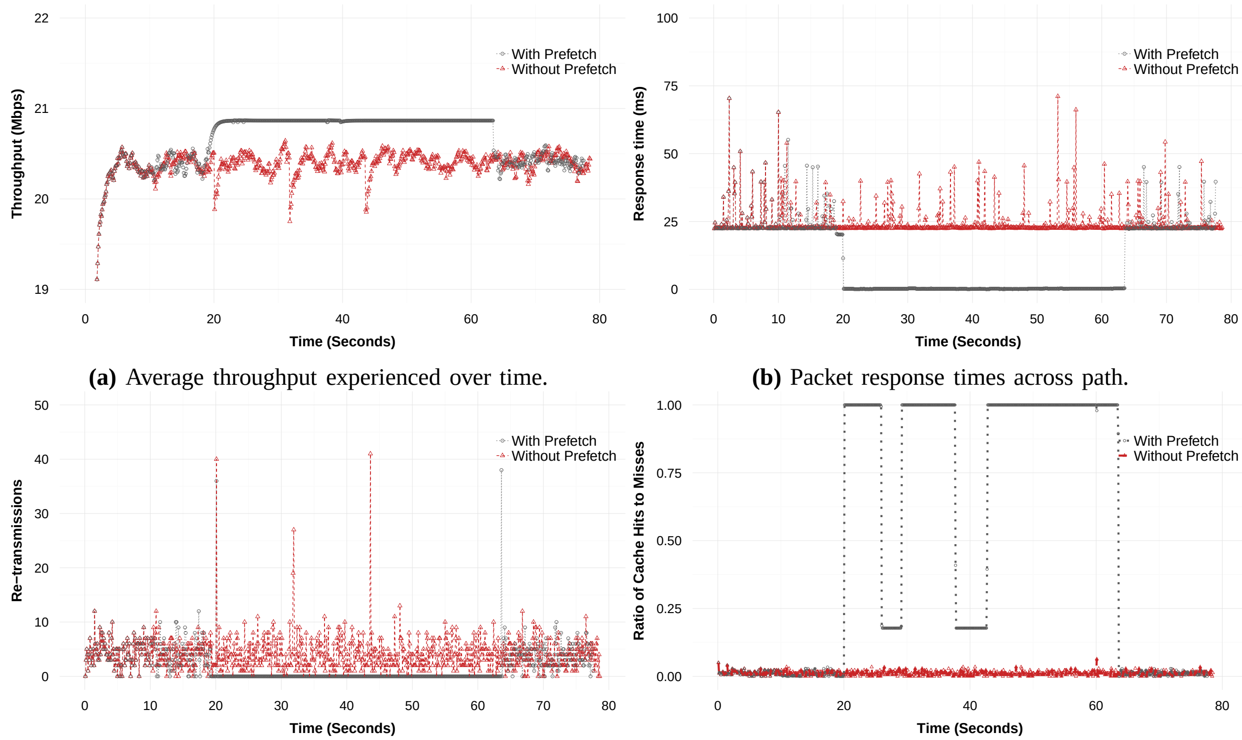
Input:
$P_N = \{(C_1, R_1, d_1, s_1), ..., (C_N, R_N, d_N, s_N)\}$
$M =$ content manifest file
$F =$ content size from manifest file
Start:
let $S_{max} = \lceil \frac{F}{MTU} \rceil$
let $S_{cur} = 1$
let $i =$ index of current cloudlet
let $j =$ index of candidate cloudlet
let $d_c =$ distance to candidate cloudlet
while $j \leq N$ **do**
  if $d_c < \Delta_0$ & $F > 0$ **then**
    Get 4-tuple P_i and P_j
    Calculate content chunk to prefetch
    let $E[D_i] = \frac{d_i}{s_i} \times R_i$
    let $E[D_p] = \frac{d_j}{s_j} \times R_j$
    let $S_{cur} = \lfloor \frac{E[D_i]}{MTU} \rfloor + S_{cur}$
    if j==N **then**
      $C_j \rightarrow$ **PreFetch**(M, S_{cur}, $\lfloor \frac{F}{MTU} \rfloor$)
      $E[D_p] = F$
    else
      $C_j \rightarrow$ **PreFetch**(M, S_{cur}, $\lfloor \frac{E[D_p]}{MTU} \rfloor$)
    end if
    $F = F - E[D_p]$
    $i = i + 1$
    $j = j + 1$
  end if
end while

S_{max}, based on the maximum transmission unit (MTU) of the network. The mobile user also maintains a current sequence number, S_{cur}, which is used to inform candidate cloudlets of the starting sequence number to begin prefetching at as the user moves across a path. Each cloudlet is equipped with a prefetching service which takes as input the content manifest file M, S_{cur}, and the expected amount to be prefetched, $E[D_p]$. Based on the mobile user's current speed, s_i, distance within the cloudlet's coverage area, d_i, and throughput, R_i, the expected amount to be downloaded within the current cloudlet is $E[D_i] = \frac{d_i}{s_i} \times R_i$. Thus, the expected sequence number for the candidate cloudlet to begin prefetching at is $S_{cur} = \lfloor \frac{E[D_i]}{MTU} \rfloor + S_{cur}$. Information from P_j is then used to calculate the content amount to be prefetched, $E[D_p]$, by the candidate cloudlet, C_j. Once a distance threshold, Δ_0, is reached the prefetching service on the candidate cloudlet is initiated. For each iteration, S_{cur} is updated based on the expected amount of content downloaded within the current cloudlet and is the starting sequence number for the candidate cloudlet's prefetching service. This process is repeated until the entire content has been prefetched or the user has arrived at its destination. Performance results of the proposed mobile content delivery scheme are discussed in Section III-B.

B. Evaluation

To evaluate the performance of the proposed mobile content delivery scheme the throughput, packet response times, number of requests re-transmitted and cache hit to cache miss ratios were measured on the simulated cluster topology shown in Figure 7. Mobile user M requests 1,800 content chunks per second from the content delivery cloudlet, $C7$, while moving across a cloudlet-aware path consisting of cloudlets $\{C_1, C_2, C_3, C_4\}$. Table IV details the simulation parameters used.

Figures 8a and 8b show the throughput and packet response times experienced by the mobile user M. The speed of the mobile user varies between 1-2 meters per second in order to simulate a brisk walk in an urban environment. Algorithm 1 is applied as the mobile user moves across its path to proactively request content in anticipation of service continuity issues and intermittent connectivity. This improves content delivery speeds and minimizes response times as requests are immediately fulfilled by the cloudlet's $L1$ caches which contain the prefetched content. Without prefetching sharp drops in throughput are visible when the mobile device associates with $C2$, $C3$ and $C4$. This can cause issues with a user's QoE especially with time sensitive content such as audio or video. Once the mobile user arrives at its destination the prefetching service is terminated. As shown in Figure 8c, prefetching also allows for the mobile device to experience virtually no re-transmissions during its movement across the path, again due to the immediate fulfillment of requests from the $L1$ caches. This is better illustrated in Figure 8d which highlights the ratio of cache hits to cache misses on the cloudlet path. Algorithm 1 ensures content chunks will be available and thus the cache hit ratio will always be 1 within the



(a) Average throughput experienced over time.

(b) Packet response times across path.

(c) Number of re-transmitted requests.

(d) Ratio of cache hits to cache misses.

Figure 8: Mobile content delivery performance results.

coverage area of candidate cloudlets. This proactive prefetching technique results in improved delivery speeds, response times and cache hit ratios providing mobile users, with intermittent connectivity, a less strenuous transition which is essential for mobile environments.

IV. Conclusion

In this paper, a detailed examination of LinkNYC's urban communications network is performed to investigate the fundamental benefits of shifting from a traditional IP network to a content-centric network where popular content is cached as it disseminates throughout the network. Traditionally, content delivery nodes are distributed globally to bring content as close to the geographic location of the user. However, promising results show that having multiple content-delivery cloudlets within LinkNYC's infrastructure dramatically improves overall network performance and stability. Finally, mobile users within LinkNYC are bound to experience intermittent connectivity as multiple cloudlets are encountered. Thus, a cloudlet-aware mobile content delivery scheme is proposed to leverage LinkNYC's infrastructure and improve a mobile user's QoE.

References

[1] B. Hamdaoui and P. Ramanathan, "Cross-layer optimized conditions for QoS support in multi-hop wireless networks with MIMO links," *Selected Areas in Communications, IEEE Journal on*, vol. 25, no. 4, 2007.

[2] B. Hamdaoui and K. G. Shin, "Characterization and analysis of multi-hop wireless mimo network throughput," in *Proc. of the 8th ACM international symposium on Mobile ad hoc networking and computing*, 2007, pp. 120–129.

[3] W. Su, A. K. Sadek, and K. R. Liu, "Cooperative communication protocols in wireless networks: performance analysis and optimum power allocation," *Wireless Per. Comm.*, vol. 44, no. 2, pp. 181–217, 2008.

[4] N. Chakchouk and B. Hamdaoui, "Uplink performance characterization and analysis of two-tier femtocell networks," *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 9, pp. 4057–4068, 2012.

[5] B. Hamdaoui and K. G. Shin, "OS-MAC: An efficient MAC protocol for spectrum-agile wireless networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 8, pp. 915–930, 2008.

[6] B. Hamdaoui, "Adaptive spectrum assessment for opportunistic access in cognitive radio networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 2, pp. 922–930, 2009.

[7] P. Venkatraman, B. Hamdaoui, and M. Guizani, "Opportunistic bandwidth sharing through reinforcement learning," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 6, pp. 3148–3153, 2010.

[8] D. Dechene and A. Shami, "Energy-aware resource allocation strategies for lte uplink with synchronous harq constraints," *IEEE Transactions on Mobile Computing*, no. 99, pp. 1–1, 2014.

[9] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, k. claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, "Named data networking," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 66–73, Jul. 2014.

[10] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14–23, Oct 2009.

[11] H. Sinky, B. Hamdaoui, and M. Guizani, "Handoff-aware cross-layer assisted multi-path TCP for proactive congestion control in mobile heterogeneous wireless networks," in *GLOBECOM 2015*, pp. 1–7.

[12] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[13] A. Afanasyev, I. Moiseenko, and L. Zhang, "ndnSIM: NDN simulator for NS-3," NDN, Technical Report NDN-0005, October 2012. [Online]. Available: http://named-data.net/techreports.html

[14] S. Mastorakis, A. Afanasyev, I. Moiseenko, and L. Zhang, "ndnSIM 2.0: A new version of the NDN simulator for NS-3," NDN, Technical Report NDN-0028, January 2015.